Estimation of Rate Control Parameters for Video Coding Using CNN

Maria Santamaria[1,*], Ebroul Izquierdo[1], Saverio Blasi[2], Marta Mrak[2]

[1] *Multimedia and Vision Group, Queen Mary University of London, London, United Kingdom*
[2] *British Broadcasting Corporation, London, United Kingdom*
[*] m.santamariagomez@qmul.ac.uk

***Abstract*—Rate-control is essential to ensure efficient video delivery. Typical rate-control algorithms rely on bit allocation strategies, to appropriately distribute bits among frames. As reference frames are essential for exploiting temporal redundancies, intra frames are usually assigned a larger portion of the available bits. In this paper, an accurate method to estimate number of bits and quality of intra frames is proposed, which can be used for bit allocation in a rate-control scheme. The algorithm is based on deep learning, where networks are trained using the original frames as inputs, while distortions and sizes of compressed frames after encoding are used as ground truths. Two approaches are proposed where either local or global distortions are predicted.**

***Index Terms*—CNN, video coding, rate-control**

I. Introduction

Modern video coding standards, such as the H.265/High Efficiency Video Coding (HEVC), make use of complex mechanisms to provide remarkable compression efficiency. For distribution, frames are encoded using so called random-access configurations, in which most frames are inter-predicted, while a few intra frames are inserted periodically in the sequence (the number of frames between two intra frames is referred to as the intra-period). Intra frame coding uses prediction to decrease spatial redundancies, transform coding of residual signals, quantisation, and entropy coding to reduce statistical redundancies [1]. Due to the inherent complexity of these modules, it is generally difficult to estimate the effects of an encoder on a given frame in terms of the number of bits and the distortion without actually encoding it. Conversely, rate-control mechanisms typically work by allocating the available number of bits per second among the frames in an intra-period, and then appropriately setting parameters to meet this allocation. Allocating the correct number of bits for intra frames is crucial, since such frames typically need significantly more bits than inter frames (due to the reduced efficiency of the encoder scheme). However, they should also be encoded at the highest quality, as they are used for reference by subsequent inter frames [2]. As such, schemes to accurately predict the number of bits and distortion generated by an intra frame encoder are highly beneficial.

A method based on deep learning to estimate distortion and number of bits needed to encode an intra frame is proposed in this paper. A first CNN is modelled to estimate the compressed frame size, measured as bits-per-pixel (bpp), and the average distortion, measured using the Peak Signal-to-Noise Ratio (PSNR) between original and compressed frames, obtained using different Quantisation Parameters (QPs). An additional CNN is also proposed to estimate distortion maps, namely pixel-wise maps of absolute differences between original and reconstructed frames, which may be used for block-wise rate-control or adaptive-quantisation schemes. The CNN computes the maps based on the original frame and an input QP.

II. Related Work

Methods based on deep learning have been shown to be very successful in different estimation tasks. In particular, Convolutional Neural Networks (CNNs) have earned a lot of attention in recent years due to their good performance, and have been extensively used for classification and segmentation [3], super resolution [4], noise removal [5] or depth estimation [6].

Deep learning has also been used in video coding for various applications, including: frame partitioning [7], intra mode selection [8], arithmetic coding [9], compressed frame sizes-distortion modelling [10] and post processing [11]. Laude and Ostermann [8] introduced a CNN-based classifier for intra mode decision. The CNN takes an input block, and outputs the predicted intra mode to be used. Training uses original samples to avoid dependencies on other encoder decisions and reconstructed data, allowing to process several blocks in parallel. Li *et al.* [7] proposed a learning-based classifier to determine the partitioning of coding tree units (CTUs). Three CNNs are modelled to learn the split decision of CTUs at different depth levels, following maximum and minimum CTU sizes on HEVC. Song *et al.* [9] introduced a two-fold CNN-based arithmetic coding. First, a CNN is used to predict the distribution of the intra modes taking as input the Most Probable Modes (MPMs) of the current block and reconstructed neighbouring blocks. Subsequently, the predicted distributions are used in a multi-level arithmetic coding engine. Zhou *et al.* [11] proposed a CNN to replace deblocking filter and Sample Adaptive Offset (SAO).

An approach was presented by Xu *et al.* [10], where CNNs are used to estimate distortion maps and compressed frame sizes. Firstly, distortion maps are calculated with respect to the Structural Similarity Index (SSIM) between the original frame and its reconstruction. Secondly, compressed frame sizes are estimated, in the form of a vector of bits obtained after encoding a frame using different QPs. Both CNNs only use linear activations and can therefore be modelled as a combination of linear functions.

III. Proposed Approach

The CNNs proposed in [10], from here on referred to as "base CNNs", were used as the starting point for the work proposed in this paper. As opposite to SSIM as used in [10], most video encoders rely on Mean Square Error (MSE) based distortions to perform encoder-side mode decisions. Additionally, due to the non-linearity of several of the encoder blocks, using only linear activations may not be sufficient to provide accurate estimates. Finally, when dealing with practical applications, there may be a need for obtaining a low-complex estimate of distortion and number of bits. As such, the approach proposed here is different from the base CNN in that it is capable of predicting MSE distortions (instead of SSIM values) and makes use of non-linear activation functions. Moreover, in addition to a methodology to obtain local distortion maps, an additional CNN is proposed here which can provide a low-complexity estimate of average distortions for the whole frame (referred to as global distortions) and number of bits for a variety of QPs, in a single pass. The estimate of such global distortions was found to be in fact more accurate than that of local distortions, as shown in the rest of this paper.

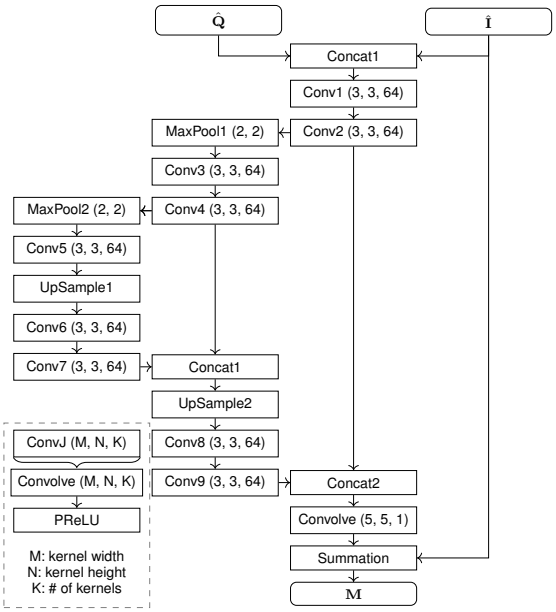

Fig. 1. CNN G. Dash-lined square indicates there are convolutional layers followed by a PReLU function.

A. Local estimation of distortion maps

The estimation of distortion maps was performed using a CNN with two inputs. The first input is the original frame data $\mathbf{I}$. Only the luminance is considered, namely a matrix of dimension $W \times H$, which is then normalised as follows:

$$\hat{I}(x,y) = \frac{I(x,y)}{2^{n-1}}, \tag{1}$$

where $x \in \{0,1,2,\ldots,W-1\}$ and $y \in \{0,1,2,\ldots,H-1\}$, n is the bitdepth of the source samples. In addition, a second input is also considered, which consists of a normalised map of QP values (with respect to the maximum QP value QP_{max}, which in HEVC is set to 51), $\hat{\mathbf{Q}}$, of dimension $W \times H$, obtained as:

$$\hat{Q}(x,y) = \frac{\text{QP}}{\text{QP}_{max}}. \tag{2}$$

For the training, a set of ground truth distortion maps $\mathbf{D}$ were used, namely sample-wise maps of absolute differences between the original and reconstructed frame. The goal of the network is to estimate the distortion map $\mathbf{M} = G(\hat{\mathbf{I}}, \hat{\mathbf{Q}}) \approx \mathbf{D}$. As shown in Fig. 1, G is an CNN formed of residual connections, convolutions, non-linear mapping, down-sampling, up-sampling and skip connections.

G initially learns the differences between inputs and outputs, where such difference is modelled in the last layer as an element-wise summation between the output of the previous layer and $\hat{\mathbf{I}}$. Secondly, convolutional layers use a stride of 1×1, and filter sizes of 3×3, except the final layer which uses a 5×5 filter. Thirdly, non-linear mapping is achieved by adding Parametric Rectified Linear Unit (PReLU) [12] after each convolutional layer, which increases the flexibility of the network. Max pooling layers adopt a filter size of 2×2, the stride is 1×1 and the output represents one quarter of the input. Up-sampling layers balance the size reduction introduced by max pooling layers. Finally, skip connections serve to aggregate multi-level features, which are modelled by concatenating the features learnt in the 2nd and 4th convolutional layers with features learnt in 9th and 7th convolutional layers, respectively.

The loss function used for training is the MSE:

$$L_G = \frac{\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}\big(D(x,y) - M(x,y)\big)^2}{W \cdot H}. \tag{3}$$

B. Estimation of number of bits and global distortions

An additional CNN was modelled to produce the estimate of the number of bits obtained with an HEVC encoder while intra coding a frame. The CNN takes as input the normalised luminance image data $\hat{\mathbf{I}}$, and is given ground truths in the form of a vector of scalars V, where each element is the number of bits necessary to encode the frame with a certain QP value. A total K QP values are considered, and therefore K is the length of the vector. The goal is to estimate the vector $P = F(\hat{\mathbf{I}}) \approx V$. As shown in Fig. 2, the mapping F is a CNN similar to G. Nevertheless, F uses Fully Connected (FC) layers that extract meaningful data from features. Moreover, convolutional layers are activated using Rectified Linear Unit (ReLU) [13], and the loss function L_F is the Mean Absolute Error (MAE):

$$L_F = \frac{\sum_{i=1}^{K}|V(i) - P(i)|}{K}. \tag{4}$$

In addition to being used for predicting the number of bits, the same CNN F was also trained to predict global average distortions. In this case, each element in the the ground truths V is mean of the distortion map between original and reconstructed frame, as obtained when encoding with a given

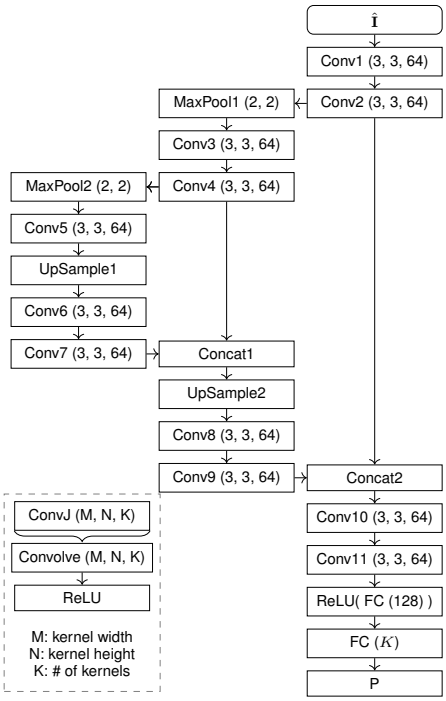

Fig. 2. CNN F. Dash-lined square indicates that each convolutional layer is followed by a ReLU function.

TABLE I
TRAINING PARAMETERS.

Batch size	Optimiser	Learning rate	Weight decay
32	Adam [14]	0.0001	0.0001

QP value.

The CNNs were trained using the parameters displayed in Table I. The stop condition was defined in terms of epochs, where an epoch is defined as a complete training obtained by feeding all available samples in the training set to the network. In particular, the training was stopped in case the validation loss did not result in any improvement after additional 10 epochs of training. Furthermore, the loss functions were regularised by adding the ℓ^2-norm of the training variables since on previous training/testing exercises better results were obtained with it.

IV. Experimental Results

The CNNs were implemented in TensorFlow and trained on an NVIDIA GeForce GTX 1080 GPU. MS COCO 2017 [15] datasets are used for running the experiments: 20,000 frames are selected for training, 5,000 for validation and 20,000 for testing. The frames are cropped into 128×128 patches and converted to YUV colour space. The HEVC reference software [16] (HM 16.9) was used. Four different QPs were considered, namely 22, 27, 32 and 37.

The proposed methods are compared with the work in [10]. The base CNNs were implemented using the description provided within [10], indicating the usage of linear activations for convolutional layers, training with Adam optimiser, learning rate of 0.001 and no regularisation. Furthermore, the training was done using a batch size of 32 and the same stop condition as in Section III was used. Additionally, the distortion is computed as the pixel-wise map of absolute differences, instead of SSIM, between original and reconstructed frames. While training the base CNNs, it was noticed that the networks would fluctuate around local minima without stabilising. This behaviour may be due to several factors, including the training dataset not being large enough or the variable updates using a too high learning rate. The proposed CNNs solve this issue by means of considering the regularisation within the loss function.

TABLE II
LOCAL CORRELATION COEFFICIENTS OF DISTORTION MAP ESTIMATES.

CNN	Region	PCC			
		QP 22	QP 27	QP 32	QP 37
Base	64×64	0.58 ± 0.5	0.57 ± 0.6	0.79 ± 0.4	0.86 ± 0.3
	32×32	0.54 ± 0.4	0.59 ± 0.4	0.71 ± 0.3	0.79 ± 0.3
	16×16	0.48 ± 0.3	0.56 ± 0.3	0.66 ± 0.3	0.76 ± 0.3
	8×8	0.41 ± 0.3	0.51 ± 0.3	0.62 ± 0.3	0.72 ± 0.2
G	64×64	0.51 ± 0.6	0.78 ± 0.4	0.89 ± 0.3	0.92 ± 0.2
	32×32	0.53 ± 0.4	0.79 ± 0.3	0.90 ± 0.2	0.92 ± 0.2
	16×16	0.54 ± 0.4	0.78 ± 0.3	0.88 ± 0.2	0.90 ± 0.2
	8×8	0.54 ± 0.3	0.76 ± 0.3	0.85 ± 0.2	0.87 ± 0.1

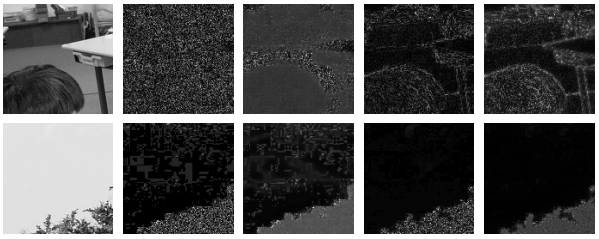

Fig. 3. Comparison of distortion maps. $\mathbf{D}$ are the ground truths and $\mathbf{M}$ are the estimates obtained using G.

Results obtained using the CNN G are presented here by measuring local correlation between the predicted $\mathbf{M}$ and real $\mathbf{D}$ distortion maps. Correlations were computed by squaring and averaging the distortion maps in blocks of different sizes. The values for each block were arranged in two vectors (one for the ground truth, and one for the estimated values, respectively), which were then compared using the Pearson Correlation Coefficient (PCC).

Table II shows a summary of the obtained PCC values in terms of QP and the size of the blocks. It can be noticed that the lower the QP, the lower the correlation between ground truths and estimates, indicating that the CNN predicts more easily in case of generically higher distortions (obtained with high QPs). Moreover, higher correlations are obtained when considering larger block sizes, which can be expected in that even in the case of local distortion estimates, the CNNs are more suitable for predicting global trends. This behaviour is confirmed through a visual comparison as exhibited in Fig. 3. Although the estimated distortion map is not capable of estimating finer details in distortion present in the ground truth,

TABLE III
COMPARISON OF COMPRESSED FRAME SIZE ESTIMATES.

Network	MAE	Fréchet distance
Base	10.454 ± 9.346	15.558 ± 16.748
F	0.067 ± 0.067	0.136 ± 0.132

TABLE IV
COMPARISON OF QUALITY ESTIMATES.

Network	MAE	Fréchet distance
Base	10.482 ± 4.710	3.577 ± 1.340
F	0.757 ± 0.719	1.260 ± 0.896
G	1.216 ± 0.359	1.635 ± 0.510

trends in distortion variation are accurately estimated. Results obtained using the CNN F are also presented both in terms of estimating global distortions and bits. These were analysed using the Fréchet distance [17] (Euclidean), which measures similarity by calculating the minimum length of leash required to connect two curves. In this case, the distance between the interpolated curve of bpp or average PSNR values over QPs obtained using ground truth and estimations was computed. Tables III and IV show these results, respectively. Average PSNR values are also reported for the G CNN.

When considering estimate of bpp values, results show that the proposed network F outperforms the base model, since lower losses and lower Fréchet distances are obtained. Fig. 4 displays bpp predictions per QP for two frames. Although difference can be seen in Fig. IV, there is a strong correlation between ground truths and predictions. Better results are obtained for higher QP values. Similarly, for distortion estimations, lower loss and lower Fréchet distance are obtained using the proposed networks. The predictions for two different frames are displayed in Fig. 5. In general, estimates obtained using F are better than those from G, confirming that global estimations may be more suitable, unless the application requires local distortions to be available.

V. CONCLUSIONS

This paper presents a CNN-based methodology to estimate distortion and number of bits obtained when intra coding original frames at different quality levels. One CNN is used to estimate vectors of compressed frame sizes or global distortions, whilst another CNN is used to estimate local distortion maps. Using the proposed methodology, these data can be estimated prior to the actual encoding process. Results show, in most cases, estimates are close and very correlated to real values. Future work includes the improvement of the CNNs, as well as the development of a complete bit allocation algorithm for rate-control applications.

ACKNOWLEDGEMENT

The work leading to this paper was co-supported by the Engineering and Physical Sciences Research Council of the UK through an iCASE grant in cooperation with the British Broadcasting Corporation and by the project COGNITUS, which received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 687605.

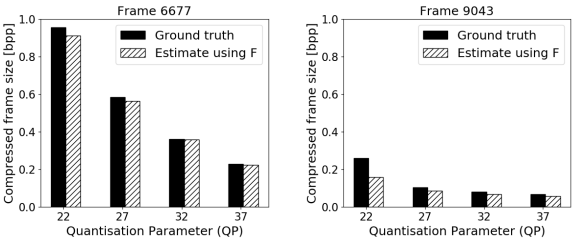

Fig. 4. Comparison of compressed frame sizes.

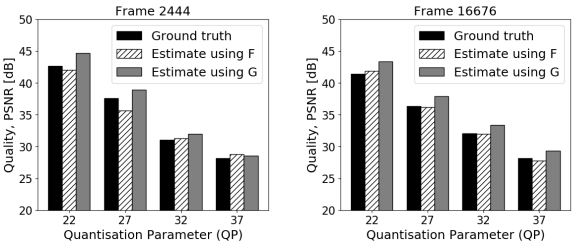

Fig. 5. Comparison of quality of reconstruction frames.

REFERENCES

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, 2012.

[2] M. Wang, K. N. Ngan, and H. Li, "An efficient frame-content based intra frame rate control for high efficiency video coding," *IEEE Signal Processing Letters*, vol. 22, no. 7, pp. 896–900, 2015.

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[4] K. K. Chua and Y. H. Tay, "Enhanced image super-resolution technique using convolutional neural network," in *Advances in Visual Informatics*, 2013, pp. 157–164.

[5] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *2013 IEEE International Conference on Computer Vision*, 2013, pp. 633–640.

[6] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, 2014, pp. 2366–2374.

[7] T. Li, M. Xu, and X. Deng, "A deep convolutional neural network approach for complexity reduction on intra-mode HEVC," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 2017.

[8] T. Laude and J. Ostermann, "Deep learning-based intra prediction mode decision for HEVC," in *2016 Picture Coding Symposium (PCS)*, 2016.

[9] R. Song, D. Liu, H. Li, and F. Wu, "Neural network-based arithmetic coding of intra prediction modes in HEVC," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017.

[10] B. Xu, X. Pan, Y. Zhou, Y. Li, D. Yang, and Z. Chen, "CNN-based rate-distortion modeling for H.265/HEVC," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017.

[11] L. Zhou, X. Song, J. Yao, L. Wang, and F. Chen, "JVET-IO022-v3: Convolutional neural network filter (CNNF) for intra frame," Tech. Rep., 2018.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1026–1034.

[13] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

[15] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Computer Vision – ECCV 2014*, 2014, pp. 740–755.

[16] Joint Collaborative Team on Video Coding (JCT-VC). HEVC test model reference software (HM). https://hevc.hhi.fraunhofer.de.

[17] T. R. Wylie, *The Discrete Frechet Distance with Applications*. Montana State University, 2013.